```
RRRRRRRRRRRR       UUU           UUU  NNN         NNN    000000000     FFFFFFFFFFFFFFF    FFFFFFFFFFFFFFF
RRRRRRRRRRRR       UUU           UUU  NNN         NNN    000000000     FFFFFFFFFFFFFFF    FFFFFFFFFFFFFFF
RRRRRRRRRRRR       UUU           UUU  NNN         NNN    000000000     FFFFFFFFFFFFFFF    FFFFFFFFFFFFFFF
RRR        RRR     UUU           UUU  NNN         NNN  000       000   FFF                FFF
RRR        RRR     UUU           UUU  NNN         NNN  000       000   FFF                FFF
RRR        RRR     UUU           UUU  NNN         NNN  000       000   FFF                FFF
RRR        RRR     UUU           UUU  NNNNNN      NNN  000       000   FFF                FFF
RRR        RRR     UUU           UUU  NNNNNN      NNN  000       000   FFF                FFF
RRR        RRR     UUU           UUU  NNNNNN      NNN  000       000   FFF                FFF
RRRRRRRRRRRR       UUU           UUU  NNN   NNN   NNN  000       000   FFFFFFFFFFF        FFFFFFFFFFF
RRRRRRRRRRRR       UUU           UUU  NNN   NNN   NNN  000       000   FFFFFFFFFFF        FFFFFFFFFFF
RRRRRRRRRRRR       UUU           UUU  NNN   NNN   NNN  000       000   FFFFFFFFFFF        FFFFFFFFFFF
RRR    RRR         UUU           UUU  NNN      NNNNNN  000       000   FFF                FFF
RRR    RRR         UUU           UUU  NNN      NNNNNN  000       000   FFF                FFF
RRR    RRR         UUU           UUU  NNN         NNN  000       000   FFF                FFF
RRR        RRR     UUU           UUU  NNN         NNN  000       000   FFF                FFF
RRR        RRR     UUU           UUU  NNN         NNN  000       000   FFF                FFF
RRR        RRR     UUUUUUUUUUUUUUUUU  NNN         NNN  000000000       FFF                FFF
RRR        RRR     UUUUUUUUUUUUUUUUU  NNN         NNH  000000000       FFF                FFF
RRR        RRR     UUUUUUUUUUUUUUUUU  NNN         NNN  000000000       FFF                FFF
```

```
CCCCCCC    000000    NN      NN  VV      VV  BBBBBBBB   BBBBBBBB
CCCCCCC    000000    NN      NN  VV      VV  BBBBBBBB   BBBBBBBB
CC         00    00  NN      NN  VV      VV  BB     BB  BB     BB
CC         00    00  NN      NN  VV      VV  BB     BB  BB     BB
CC         00    00  NNNN    NN  VV      VV  BB     BB  BB     BB
CC         00    00  NNNN    NN  VV      VV  BB     BB  BB     BB
CC         00    00  NN  NN  NN  VV      VV  BBBBBBBB   BBBBBBBB
CC         00    00  NN  NN  NN  VV      VV  BBBBBBBB   BBBBBBBB
CC         00    00  NN    NNNN  VV      VV  BB     BB  BB     BB
CC         00    00  NN    NNNN  VV      VV  BB     BB  BB     BB
CC         00    00  NN      NN   VV    VV   BB     BB  BB     BB    ....
CC         00    00  NN      NN   VV    VV   BB     BB  BB     BB    ....
CCCCCCC    000000    NN      NN    VV  VV    BBBBBBBB   BBBBBBBB     ....
CCCCCCC    000000    NN      NN     VV       BBBBBBBB   BBBBBBBB     ....


LL         IIIIII       SSSSSSSS
LL         IIIIII       SSSSSSSS
LL           II       SS
LL           II       SS
LL           II       SS
LL           II       SS
LL           II          SSSSSS
LL           II          SSSSSS
LL           II                SS
LL           II                SS
LL           II                SS
LL           II                SS
LLLLLLLLLL  IIIIII    SSSSSSSS
LLLLLLLLLL  IIIIII    SSSSSSSS
```

```
  1   0001  0  MODULE CONVBB (                                         !
  2   0002  0                    IDENT = 'V04-000'
  3   0003  0  %BLISS32[,
  4   0004  0         ADDRESSING_MODE(EXTERNAL=LONG_RELATIVE,NONEXTERNAL=LONG_RELATIVE)
  5   0005  0         ]
  6   0006  0                  ) =
  7   0007  1  BEGIN
  8   0008  1  !
  9   0009  1  !*****************************************************************************
 10   0010  1  !*                                                                           *
 11   0011  1  !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
 12   0012  1  !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
 13   0013  1  !*  ALL RIGHTS RESERVED.                                                     *
 14   0014  1  !*                                                                           *
 15   0015  1  !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
 16   0016  1  !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
 17   0017  1  !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
 18   0018  1  !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
 19   0019  1  !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
 20   0020  1  !*  TRANSFERRED.                                                             *
 21   0021  1  !*                                                                           *
 22   0022  1  !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
 23   0023  1  !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
 24   0024  1  !*  CORPORATION.                                                             *
 25   0025  1  !*                                                                           *
 26   0026  1  !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
 27   0027  1  !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
 28   0028  1  !*                                                                           *
 29   0029  1  !*                                                                           *
 30   0030  1  !*****************************************************************************
 31   0031  1
 32   0032  1  !++
 33   0033  1  ! FACILITY:      DSR (Digital Standard RUNOFF) / DSRPLUS
 34   0034  1  !
 35   0035  1  ! ABSTRACT: Convert a binary number into a vector of characters and
 36   0036  1  !           return the result and character count.
 37   0037  1  !
 38   0038  1  !
 39   0039  1  ! ENVIRONMENT: Transportable
 40   0040  1  !
 41   0041  1  ! AUTHOR: R.W.Friday     CREATION DATE: May, 1979
 42   0042  1  !
```

```
:   44      0043  1 %SBTTL 'Revision History'
:   45      0044  1 !
:   46      0045  1 ! MODIFIED BY:
:   47      0046  1 !
:   48      0047  1 !     002     KFA00002       Ken Alden        07-Mar-1983
:   49      0048  1 !             Global edit of all modules. Updated module names, idents,
:   50      0049  1 !             copyright dates. Changed require files to BLISS library.
:   51      0050  1 !
:   52      0051  1 !--
```

CONVBB
V04-000

Module Level Declarations

I 15
16-Sep-1984 00:10:31   VAX-11 Bliss-32 V4.0-742
14-Sep-1984 13:05:52   [RUNOFF.SRC]CONVBB.BLI;1

Page 3
(3)

```
:  54    0052  1 %SBTTL 'Module Level Declarations'
:  55    0053  1
:  56    0054  1 !
:  57    0055  1
```

```
59    0056  1  GLOBAL ROUTINE CONVBB (BINARY_NUMBER, KHARACTERS, KHARACTER_COUNT, BASE) : NOVALUE =
60    0057  1
61    0058  1  !++
62    0059  1  ! FUNCTIONAL DESCRIPTION:
63    0060  1  !
64    0061  1  !     Converts 'binary_number' to a vector of characters,
65    0062  1  !     returning them in 'kharacters'; kharacter_count is the
66    0063  1  !     number of digits converted.
67    0064  1  !     The absolute value of 'binary_number' is converted,
68    0065  1  !     so that the user is responsible for handling negative numbers.
69    0066  1  !     The number will be converted according to the value of BASE.
70    0067  1  !
71    0068  1  ! FORMAL PARAMETERS:
72    0069  1  !
73    0070  1  !     See FUNCTIONAL DESCRIPTION
74    0071  1  !
75    0072  1  ! IMPLICIT INPUTS:
76    0073  1  !
77    0074  1  !     NONE
78    0075  1  !
79    0076  1  ! IMPLICIT OUTPUTS:
80    0077  1  !
81    0078  1  !     NONE
82    0079  1  !
83    0080  1  ! ROUTINE VALUE:
84    0081  1  ! COMPLETION CODES:
85    0082  1  !
86    0083  1  !     NONE
87    0084  1  !
88    0085  1  ! SIDE EFFECTS:
89    0086  1  !
90    0087  1  !     NONE
91    0088  1  !
92    0089  1  !--
93    0090  1
94    0091  2      BEGIN
95    0092  2
96    0093  2      OWN
97    0094  2          DIGITS : INITIAL (CH$PTR(UPLIT('0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ')));
98    0095  2
99    0096  2      MAP
100   0097  2          KHARACTERS : REF VECTOR;
101   0098  2
102   0099  2      LOCAL
103   0100  2          LEFT_TO_CONVERT;
104   0101  2
105   0102  2      .KHARACTER_COUNT = 0;
106   0103  2      LEFT_TO_CONVERT = ABS (.BINARY_NUMBER);
107   0104  2
108   0105  2      DO
109   0106  3          BEGIN
110   0107  3          KHARACTERS [..KHARACTER_COUNT] = CH$RCHAR( CH$PLUS(.DIGITS, (.LEFT_TO_CONVERT MOD .BASE)));
111   0108  3          LEFT_TO_CONVERT = .LEFT_TO_CONVERT/.BASE;
112   0109  3          .KHARACTER_COUNT = ..KHARACTER_COUNT + 1;
113   0110  3          END
114   0111  2      UNTIL .LEFT_TO_CONVERT EQL 0;
115   0112  2
```

```
; 116          0113 2     RETURN;
; 117          0114 1     END;                                    !End of CONVBB

                                          .TITLE   CONVBB
                                          .IDENT   \V04-000\

                                          .PSECT   $SPLIT$,NOWRT,NOEXE,2

45 44 43 42 41 39 38 37 36 35 34 33 32 31 30 00000 P.AAA: .ASCII  \0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ\
54 53 52 51 50 4F 4E 4D 4C 4B 4A 49 48 47 46 0000F
                              5A 59 58 57 56 55 0001E

                                          .PSECT   $OWN$,NOEXE,2

                    00000000' 00000 DIGITS: .ADDRESS P.AAA


                                          .PSECT   $CODE$,NOWRT,2

                                 0004 00000   .ENTRY  CONVBB, Save R2
                        0C BC D4 00002        CLRL    @KHARACTER_COUNT
                  52 04 AC D0 00005           MOVL    BINARY_NUMBER, LEFT_TO_CONVERT
                        03 18 00009           BGEQ    1$
                  52 52 CE 0000B              MNEGL   LEFT_TO_CONVERT, LEFT_TO_CONVERT
               51 0C BC D0 0000E 1$:          MOVL    @KHARACTER_COUNT, R1
  7E       00        52 01 7A 00012           EMUL    #1, LEFT_TO_CONVERT, #0, -(SP)
  50       50        8E 10 AC 7B 00017        EDIV    BASE, (SP)+, R0, R0
08 BC41 00000000'FF40 9A 0001D                MOVZBL  @DIGITS[R0], @KHARACTERS[R1]
                  52 10 AC C6 00027           DIVL2   BASE, LEFT_TO_CONVERT
                     0C BC D6 0002B           INCL    @KHARACTER_COUNT
                        52 D5 0002E           TSTL    LEFT_TO_CONVERT
                        DC 12 00030           BNEQ    1$
                           04 00032           RET

; Routine Size: 51 bytes,   Routine Base: $CODE$ + 0000

; 118          0115 1
; 119          0116 1 END                                        !End of module
; 120          0117 0 ELUDOM
```

CONVBB
V04-000   Module Level Declarations

L 15
16-Sep-1984 00:10:31   VAX-11 Bliss-32 V4.0-742
14-Sep-1984 13:05:52   [RUNOFF.SRC]CONVBB.BLI;1

Page   6
      (4)

```
;                            COMMAND QUALIFIERS

;        BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:CONVBB/OBJ=OBJ$:CONVBB MSRC$:CONVBB/UPDATE=(ENH$:CONVBB)

; Size:          51 code + 40 data bytes
; Run Time:         00:01.7
; Elapsed Time:     00:05.7
; Lines/CPU Min:    4153
; Lexemes/CPU-Min:  9230
; Memory Used: 23 pages
; Compilation Complete
```